

# **Integration of Health Maintenance Guidelines Into a Clinical Workstation Using Expert System And Relational Database Technology**

**Robert A. Jenders, MD and G. Octo Barnett, MD**

**Laboratory of Computer Science, Massachusetts General Hospital  
Boston, Massachusetts**

## **ABSTRACT**

*We are developing a clinical workstation which integrates access to health maintenance guidelines with a computer-based medical record. We discuss the use of relational database and expert system technology to provide both patient-specific and patient-independent access to clinical guidelines.*

## **SYSTEM PLATFORM**

The Clinical Workstation (CWS) resides on a 486-based computer and is constructed in the Windows™-based ToolBook® software package. In the transitional version, permanent patient data are stored in a remote COSTAR™ system. These data are loaded into an Oracle® relational database on another remote server prior to a patient visit. The CWS then uses the DDE protocol and embedded Structured Query Language (SQL) commands to retrieve data when a patient's record is requested by a user. A number of workstations will be installed, each linked to a single central server which holds patient data and the expert system for health maintenance guidelines.

## **IMPLEMENTATION OF PATIENT-SPECIFIC GUIDELINES DISPLAY**

For the initial CWS, we chose guidelines whose logical rules depend only on coded data in the patient record and thus require no user interaction for processing. These include guidelines for screening mammography, clinical breast exam, fecal occult blood and administration of pneumovax, influenza vaccine and tetanus toxoid. Each of these guidelines was written as a set of rules or medical logic module (MLM) in the Arden Syntax. The MLMs formed a paper-based reference library for implementation.

Each MLM then was codified as a separate knowledge base (KB) in the NEXPERT OBJECT™ expert system shell using a built-in graphical editor. This software includes a specialized bridge for an Oracle database, so SQL commands which retrieve patient data for processing the MLM rules may be embedded in the KB. Since these rules require no

user interaction to process, the NEXPERT server executes separately from the CWS at the time a patient record is loaded into Oracle® in anticipation of a scheduled clinic visit.

Each MLM makes a recommendation either for or against performing a particular health maintenance intervention. This conclusion is stored in an Oracle® table which is queried directly by the CWS when a user accesses a particular patient record. In addition, each condition in each MLM is associated with text strings which explain the condition depending on its truth value. Those valid strings supporting the MLM's recommendation are concatenated and stored with the recommendation in the Oracle table. The CWS at a user's request then can use embedded SQL to extract this text string for interactive display. Thus, the CWS provides patient-specific justification for its recommendations.

## **IMPLEMENTATION OF PATIENT- INDEPENDENT GUIDELINES DISPLAY**

The static text of each guideline, together with supporting literature references, is stored in another Oracle® table. We have implemented a "browser" in the CWS which displays the text for each guideline without accessing a particular patient's record. In addition, this static text is combined with the patient-specific explanatory text strings to provide a complete explanation facility to justify guideline recommendations.

## **CONCLUSIONS**

The use of relational database technology offers a facile way to store and retrieve both structured patient data and health maintenance recommendations based on those data. This provides a point of communication between a workstation and a remote expert system. Such technology affords standard query mechanisms and centralized data structures, making database maintenance easier. Also, a commercial expert system shell minimizes the burden of procedural programming while providing access to a structured patient record stored in a relational database.